Package ‘rjwsacruncher’

February 7, 2025

Title Interface to the 'JWSACruncher' of 'JDemetra+'

Version 0.2.1

Description 'JDemetra+' (<https://github.com/jdemetra/jdemetra-app>) is the seasonal adjustment software officially recommended to the members of the European Statistical System and the European System of Central Banks. Seasonal adjustment models performed with 'JDemetra+' can be stored into workspaces. 'JWSACruncher' (<https://github.com/jdemetra/jwsacruncher/releases>) is a console tool that re-estimates all the multi-processing defined in a workspace and to export the result. 'rjwsacruncher' allows to launch easily the 'JWSACruncher'.

URL https://github.com/AQLT/rjwsacruncher

BugReports https://github.com/AQLT/rjwsacruncher/issues

Imports XML

Suggests knitr, rmarkdown

License GPL-3

VignetteBuilder knitr

Encoding UTF-8

RoxygenNote 7.3.2

NeedsCompilation no

Author Alain Quartier-la-Tente [aut, cre]
(<https://orcid.org/0000-0001-7890-3857>),
Institut national de la statistique et des études économiques [cph]

Maintainer Alain Quartier-la-Tente <alain.quartier@yahoo.fr>

Repository CRAN

Date/Publication 2025-02-07 15:10:02 UTC

Contents

`configure_jwsacruncher` *Configure the 'JWSACruncher' with a portable version of 'Java'*

Description

Function configure the 'JWSACruncher' with a portable version of 'Java'.

Usage

```
configure_jwsacruncher(jwsacruncher_path, java_path)
```

Arguments

`jwsacruncher_path`
Path to the file `jwsacruncher.bat` of the 'JWSACruncher' (see details).

`java_path` Path to the file `java.exe` of the portable version of 'Java' (see details).

Details

Since the version 2.2.0, the 'JWSACruncher' needs 'Java' 8 or higher to run. For versions 3.0.0 and higher, 'JWSACruncher' needs 'Java' 17 or higher. In 'Windows' versions 3.0.0 and higher of 'JWSACruncher' includes a portable version of 'Java'. For lower version of 'JWSACruncher', if you cannot install 'Java' (for example for security reasons) you can install a portable version of 'Java' (that does not require administrator rights) and configure the 'JWSACruncher' to use this portable version. To do it you have to:

1. Unzip the downloaded file of the 'JWSACruncher';
2. Open, with a Text Editor, the file `jwsacruncher.bat` that is in the sub-folder `/bin/` of the 'JWSACruncher';
3. Edit the line 71 that contains `if "%JAVACMD%"=="" set JAVACMD=java` and replace `java` by the path to the file `java.exe` of the portable version. For example, if the portable version of 'Java' is installed under `D:/Software/Java`, the path to `java.exe` should be at `D:/Software/Java/bin/java.exe` and the new line 71 would be `if "%JAVACMD%"=="" set JAVACMD="D:\Software\Java\bin\java.exe"`.

The function `configure_jwsacruncher()` does the steps 2 and 3.

See Also

download_cruncher().

create_param_file *Create parameter file for the 'JWSACruncher'*

Description

To run the 'JWSACruncher' needs a parameter file and `create_param_file` allows to create it.

Usage

```
create_param_file(
  dir_file_param,
  bundle = 10000,
  csv_layout = "list",
  csv_separator = ";",
  ndecs = 6,
  full_series_name = TRUE,
  rslt_name_level = 2,
  policy = "parameters",
  refreshall = TRUE,
  output = NULL,
  matrix_item = getOption("default_matrix_item"),
  tsmatrix_series = getOption("default_tsmatrix_series"),
  paths_path = NULL,
  v3 = getOption("is_cruncher_v3"),
  file_param = file.path(dir_file_param, "parameters.param")
)
```

Arguments

`dir_file_param` Path to the directory that will contains the parameter file "`parameters.param`" (if `file_param` not supplied).

`bundle` Maximum size for a group of series (in output). By default `bundle = 10000`.

`csv_layout` Layout of the CSV files (series only). By default `csv_layout = "list"`. Other options: `csv_layout = "vtable"` (vertical table) or `csv_layout = "htable"` (horizontal table).

`csv_separator` The field separator string used in the CSV file. By default `csv_separator = ";"`.

`ndecs` Number of decimals used in the output. By default `ndec = 6`.

`full_series_name`
Boolean indicating if the fully qualified name of the series will be used (the default `full_series_name = TRUE`) or if only the name of the series should be displayed. Only used when `v3 = TRUE`.

`rslt_name_level`
Only used when `v3 = TRUE`.

`policy`
Refreshing policy of the processing. By default `policy = "parameters"` (re-estimation of the coefficients of the reg-ARIMA model, see details).

`refreshall`
Boolean indicating if the data is refreshed (by default `refreshall = TRUE`).

`output`
Full path of the output folder. By default (`output = NULL`) a folder is create in the path to the workspace ([workspace]/Output).

`matrix_item`
Character containing the items of the matrix output (see the 'JDemetra+' manual for more information). By default, the items defined in the option `getOption("default_matrix_item")` are used (option initialized by the default output of the 'JWSACruncher' 2.2.2).

`tsmatrix_series`
Character containing the names of the times series to export (see the 'JDemetra+' manual for more information). By default, the items defined in the option `getOption("default_tsmatrix_series")` are used (option initialized by the default output of the 'JWSACruncher' 2.2.2).

`paths_path`
The paths used for relative addresses (see the "Demetra Paths" of the graphical interface of 'JDemetra+').

`v3`
Boolean indicating if the parameter file should be compatible with a version 3.0.0 and higher of 'JWSACRuncher' (`v3 = TRUE`) or a lower version (`v3 = FALSE`). By default the value of the option `"is_cruncher_v3"` is used (equals to `FALSE` by default).

`file_param`
Path to the parameters file. By default the file is named `parameters.param` and it is created at the `fir_file_param` directory.

Details

When the 'JWSACruncher' is launched, the data is refreshed with a specific policy that is defined by the parameter `policy`. The available options are:

- `policy = "current"`: all the estimations are fixed and AO added for new data (since v.2.2.3), short name `policy = "n"`;
- `policy = "fixed"`: all the estimations are fixed (since v.2.2.3), short name `policy = "f"`;
- `policy = "fixedparameters"`: re-estimation of the coefficients of the regression variables (but not the ARIMA coefficients), short name `policy = "fp"`;
- `policy = "fixedarparameters"`: re-estimation of the coefficients of the regression variables and of the MA coefficients of the ARIMA model (but not the AR coefficients), short name `policy = "farp"` (since v.3.3.1);
- `policy = "parameters"` (the default): `policy = "fixedparameters"` + re-estimation of ARIMA coefficients, short name `policy = "p"`;
- `policy = "lastoutliers"`: `policy = "parameters"` + re-identification of last outliers (on the last year), short name `policy = "l"`;
- `policy = "outliers"`: `policy = "lastoutliers"` + re-identification of all outliers, short name `policy = "o"`;
- `policy = "stochastic"`: `policy = "outliers"` + re-identification of ARIMA orders, short name `policy = "s"`;
- `policy = "complete"` or `policy = "concurrent"`: the model is completely re-identified and re-estimated, short name `policy = "c"`.

Value

Path to the parameter file.

See Also

read_param_file(), list2param_file(), default_param_file cruncher_and_param().

Examples

```
## Not run:
dir = tempdir()
# Here a file parameters.param is created in the directory dir
# with default parameters of the different options
create_param_file(dir)
# to only export the raw and the seasonally adjusted series
 create_param_file(dir,
                   tsmatrix_series = c("y", "sa"))

## End(Not run)
```

---

cruncher *Run the 'JWSACruncher'*

Description

Function to run the 'JWSACruncher' on a workspace from a parameter file.

Usage

```
cruncher(
  workspace,
  cruncher_bin_directory = getOption("cruncher_bin_directory"),
  param_file_path,
  log_file = NULL,
  rename_multi_documents = FALSE,
  delete_existing_file = FALSE
)
```

Arguments

`workspace` Path to the workspace. By default a dialog box opens to choose the workspace.

`cruncher_bin_directory`
Path to the "bin" directory of the 'JWSACruncher'. By default the value of the option "cruncher_bin_directory" is used.

`param_file_path`
Path to the parameter file of the 'JWSACruncher'. By default a .params file is search in the save directory of the workspace.

`log_file` Name of the log file of 'JWSACruncher'. By default the log isn't exported.

`rename_multi_documents`
Boolean indicating whether to rename the folders containing the outputs according to the names of the multi-documents of the workspace. By default `rename_multi_documents = FALSE`: the names of the XML files of the multi-documents are used.

`delete_existing_file`
Only used if `rename_multi_documents = TRUE`. Boolean indicating whether to delete existing folders when renaming them. By default (`delete_existing_file = FALSE`) they are not deleted.

Value

The path to the workspace.

See Also

Around the 'JWSACruncher': `cruncher_and_param()`, `update_workspace()`. To create the parameter file: `create_param_file()` and `list2param_file()`.

Examples

```
## Not run:
dir = tempdir()
# First create a parameter file
# here a file parameters.param is created in the directory dir
# with default parameters of 'JWSACruncher' v2
list2param_file(dir, default_param_file(v3 = FALSE))
# If the option "cruncher_bin_directory" is correctly defined:
cruncher("workspace.xml",
          param_file_path = file.path(dir, "parameters.param"))

## End(Not run)
```

cruncher_and_param *Run quickly the 'JWSACruncher'*

Description

Function to run the 'JWSACruncher' on a workspace while creating the parameter file.

Usage

```
cruncher_and_param(
  workspace = NULL,
  output = NULL,
  rename_multi_documents = FALSE,
  delete_existing_file = FALSE,
  log_file = NULL,
  cruncher_bin_directory = getOption("cruncher_bin_directory"),
  ...
)
```

Arguments

`workspace` Path to the workspace. By default a dialog box opens to choose the workspace.

`output` Full path of the output folder. By default (`output` = `NULL`) a folder is create in the path to the workspace ([workspace]/Output).

`rename_multi_documents`
Boolean indicating whether to rename the folders containing the outputs according to the names of the multi-documents of the workspace. By default `rename_multi_documents` = `FALSE`: the names of the XML files of the multi-documents are used.

`delete_existing_file`
Only used if `rename_multi_documents` = `TRUE`. Boolean indicating whether to delete existing folders when renaming them. By default (`delete_existing_file` = `FALSE`) they are not deleted.

`log_file` Name of the log file of 'JWSACruncher'. By default the log isn't exported.

`cruncher_bin_directory`
Path to the "bin" directory of the 'JWSACruncher'. By default the value of the option `"cruncher_bin_directory"` is used.

`...` Other parameters of `create_param_file()`.

Value

Path to the workspace.

See Also

`cruncher()`, `update_workspace()`, `create_param_file()`, `multiprocessing_names()`.

`default_param_file` *Get default parameters of the 'JWSACruncher'*

Description

Get default parameters of the 'JWSACruncher'

Usage

```
default_param_file(
  v3 = getOption("is_cruncher_v3"),
  cruncher_bin_directory = NULL
)
```

Arguments

`v3` Boolean indicating if the parameters are the from a version 3.0.0 and higher of 'JWSACRuncher' (`v3 = TRUE`) or a lower version (`v3 = FALSE`). By default the value of the option `"is_cruncher_v3"` is used (equals to `FALSE` by default).

`cruncher_bin_directory` Path to the directory that contains the 'JWSACruncher' binary. If defined, the parameter `v3` is ignored and the 'JWSACruncher' is run without parameter to generate the default parameters file.

See Also

`create_param_file()`, `read_param_file()`, `list2param_file()`, `cruncher_and_param()`.

download_cruncher *Dowload the 'JWSACruncher'*

Description

Function to download the ZIP file of the 'JWSACruncher'

Usage

```
download_cruncher(
  directory,
  cruncher_version,
  v3 = getOption("is_cruncher_v3")
)
```

Arguments

`directory` Directory where to save the 'JWSACruncher'. In Windows, a dialog box opens by default to select the directory.

`cruncher_version` Character of the version of the 'JWSACruncher' to download ("X.Y.Z" format). By default the last version is downloaded.

`v3` Boolean indicating, when parameter `cruncher_version` is missing, if the last version of the 'JWSACruncher' should be a 3.x.y version or a 2.x.y. By default the value of the option `"is_cruncher_v3"` is used (equals to `FALSE` by default).

Details

The 'JWSACruncher' is downloaded from https://github.com/jdemetra/jwsacruncher/releases for versions lower than 3.0.0 and from https://github.com/jdemetra/jdplus-main/releases for the other versions. To use it, it has to be unzip.

See Also

`configure_jwsacruncher()`.

Examples

```
## Not run:
# On Windows opens a dialog box to choose the directory where to
# download the last version of the 'JWSACruncher'
download_cruncher()

dir <- tempdir()
# To download the last release:
download_cruncher(dir)

# To download the release of the version 2.2.2:
download_cruncher(dir, "2.2.2")

## End(Not run)
```

`list2param_file` *Create parameter file from list*

Description

Create parameter file from list

Usage

```
list2param_file(
  dir_file_param,
  x,
  file_param = file.path(dir_file_param, "parameters.param")
)
```

Arguments

`dir_file_param` Path to the directory that will contains the parameter file `"parameters.param"` (if `file_param` not supplied).

`x` A list, for example create by `read_param_file()` or `default_param_file()`.

`file_param` Path to the parameters file. By default the file is named `parameters.param` and it is created at the `fir_file_param` directory.

See Also

`create_param_file()`, `read_param_file()`, `default_param_file()` `cruncher_and_param()`.

Examples

```
## Not run:
dir = tempdir()
# Here a file parameters.param is created in the directory dir
# with default parameters of 'JWSACruncher' v2
list_param <- default_param_file(v3 = FALSE)
list2param_file(dir, list_param)
# to only export the raw and the seasonally adjusted series
list_param$tsmatrix_series <- c("y", "sa")
list2param_file(dir, list_param)

## End(Not run)
```

multiprocessing_names *Get the names of the multiprocessings of a workspace*

Description

Function to get the name of the multiprocessings that appears on 'JDemetra+' and the name of the corresponding XML file.

Usage

```
multiprocessing_names(workspace)
```

Arguments

workspace	Path to the workspace. By default a dialog box opens to choose the workspace.

Value

A data.frame containing the name of the multiprocessings that appears on 'JDemetra+' (column name) and the name of the associated XML files (column file).

See Also

cruncher_and_param().

read_param_file *Read parameter file of the 'JWSACruncher'*

Description

Read parameter file of the 'JWSACruncher'

Usage

```
read_param_file(file)
```

Arguments

`file`	Path to the parameter file.

See Also

`create_param_file()`, `list2param_file()`, `default_param_file()` `cruncher_and_param()`.

Examples

```
dir = tempdir()
list_param <- default_param_file(v3 = FALSE)
list2param_file(dir, list_param)
list_param_2 <- read_param_file(file.path(dir, "parameters.param"))
all.equal(list_param, list_param_2)
```

update_workspace *Update a workspace*

Description

Function to update a workspace without exporting the results.

Usage

```
update_workspace(
  workspace = NULL,
  policy = "parameters",
  cruncher_bin_directory = getOption("cruncher_bin_directory"),
  log_file = NULL
)
```

Arguments

`workspace` Path to the workspace. By default a dialog box opens to choose the workspace.

`policy` Refreshing policy of the processing. By default `policy = "parameters"` (re-estimation of the coefficients of the reg-ARIMA model, see details).

`cruncher_bin_directory`
Path to the "bin" directory of the 'JWSACruncher'. By default the value of the option `"cruncher_bin_directory"` is used.

`log_file` Name of the log file of 'JWSACruncher'. By default the log isn't exported.

Value

Path to the workspace.

See Also

`cruncher()`, `cruncher_and_param()`.

Index